
OpenICE-SLAM: Open-Set Neural Implicit Encoding for Semantic SLAM

Gianluca Sabatini*
ETH Zürich
gsabatini@ethz.ch

Isotta Magistrali*
ETH Zürich
imagistrali@ethz.ch

Gabriele Mura*
ETH Zürich
gamura@ethz.ch

Lorenzo Venturoli*
ETH Zürich
lventuroli@ethz.ch

Johanna Wald
Google
johannawald@google.com

Daniel Barath
ETH Zürich, Google
dbarath@ethz.ch

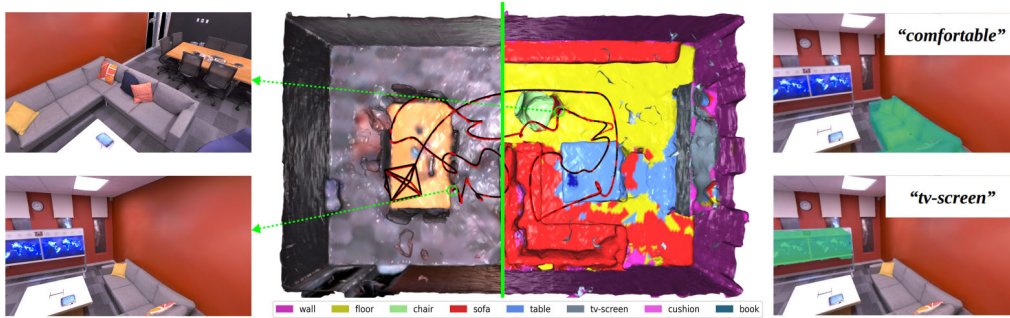


Figure 1: **Open-vocabulary 3D semantic segmentation** on a mesh from the Replica dataset (25) using the proposed OpenICE-SLAM framework, which jointly estimates scene geometry, camera trajectory, and open-set semantics. Semantic predictions are visualized on the mesh by projecting them to closed-set labels for clarity. On the right, we demonstrate free-text querying by highlighting regions corresponding to the queried terms.

Abstract

We propose a novel neural implicit SLAM framework, OpenICE-SLAM, that jointly reconstructs geometry, appearance, and open-set semantic labels in an on-line manner. Our approach introduces an implicit semantic field that continuously predicts open-set semantic features alongside geometry and appearance, enabling comprehensive scene understanding during camera tracking. Crucially, we incorporate a transformer-based module that leverages the reconstructed geometry to refine the implicit semantic field, promoting consistent and open-set semantic predictions by relating features at neighboring 3D locations. This context-aware fusion enhances temporal and spatial consistency while improving the accuracy and coherence of semantics. Unlike previous neural SLAM methods, our pipeline is the first to achieve online mapping that integrates geometry and open-set semantics via learned, context-sensitive semantic representations, enabling robust semantic understanding across diverse object classes. The code will be open-source.

1 Introduction

Simultaneous Localization and Mapping is a fundamental building block for numerous real-time applications in robotics, augmented reality, and autonomous systems. SLAM enables an agent to

*Equal contribution. Listing order is random.

incrementally reconstruct an unknown environment while simultaneously tracking its location within this map. In many applications, particularly those set up in complex and dynamic real-world scenarios, SLAM systems are required to construct a dense map, often utilizing RGB-D sensors. Among the various scene representations proposed over the years, implicit encodings have emerged as a strong baseline, with promising recent developments like NICE-SLAM (40) highlighting the potential of neural implicit representations for robust and high-quality dense mapping.

While state-of-the-art dense SLAM systems (40) manage to reliably reconstruct the geometric structure of an environment, geometry alone is often not sufficient for complex applications. Intelligent agent interactions require a comprehensive scene understanding. Thus, several prior works have integrated semantics into different SLAM systems (36; 17; 8), typically relying on a closed-set of object classes. While effective in controlled environments, real-world environments are unpredictable and diverse. An adaptable method can categorize a broad range of semantic classes, extending beyond a limited, predetermined class set.

This growing need for rich and adaptable scene understanding led to the emergence of open-vocabulary scene understanding (23; 29). Unlike conventional, closed-vocabulary semantic segmentation, open-vocabulary scene understanding can identify and interpret novel object categories using natural language queries, extending beyond a fixed taxonomy. This is made possible through cross-modal learning, which aligns visual features (e.g., from images) and textual information (e.g., from captions) into a joint, semantically rich embedding space. In this shared space, semantically related concepts such as the word 'chair' or 'sitting' and an image of a particular chair model – are embedded in close proximity. This alignment allows models to parse any text prompts and compare features to segment a scene using the underlying joint feature representation. Recent work has extended open-set capabilities to 3D environments (21; 29; 14; 28) by leveraging 3D geometry and multi-view consistency. However, these methods typically require either a pre-existing 3D scene and posed images and/or extensive training time to learn a 3D scene representation.

In contrast, we propose a novel online, neural open-set SLAM system. Our approach jointly optimizes camera localization, geometry, and (open-vocabulary) semantic features. Our key contribution is the ability to simultaneously learn an open-vocabulary semantic field alongside reconstructing geometry and color/appearance (see Fig. 1). This allows for efficient and robust mapping and localization while simultaneously providing fine-grained semantic understanding. Our primary contributions are:

- A dense RGB-D SLAM system that seamlessly integrates open-vocabulary segmentation within a neural implicit scene representation.
- We propose a novel graph transformer that improves semantic segmentation quality compared to our baseline.

Our method achieves results on par with the state-of-the-art offline open-vocabulary 3D segmentation methods while not requiring posed images and an existing reconstruction.

2 Related Work

Neural Implicit Representations. Neural implicit models have emerged as a powerful way to continuously represent 3D scenes. Neural Radiance Fields (NeRF) (18) pioneered the encoding of the appearance and geometry of an underlying 3D scene in a coordinate-based neural network, enabling the synthesis of novel high-fidelity views. Extensions (20; 33; 22; 22; 22; 1; 2; 3) have improved fidelity and geometric modeling, for example, by representing surfaces via signed distance or occupancy fields instead of volume density. These approaches learn a scene-specific function that maps 3D position (and viewing direction) to color or occupancy, obtaining smooth, high-resolution reconstructions beyond the limits of voxel grids, while often also being more storage-efficient. However, they typically focus on geometry and appearance, while semantic information is often limited to fixed-label predictions and, rarely, open-set semantics. Our work builds on neural fields as a scene representation, but extends them to predict *open-set semantic labels* in a SLAM setting.

SLAM with Neural Representations. Integrating neural implicit representations into SLAM has shown promise for dense mapping and localization. iMAP (27) first demonstrated that a single multi-layer perceptron can serve as the map of a SLAM system, jointly optimized with camera poses to produce a 3D reconstruction online. NICE-SLAM (40) addressed these issues by introducing a multi-level grid of latent codes decoded by local MLPs, allowing scalable and detailed reconstruction of indoor scenes. Co-SLAM (32) employs a hybrid coordinate-parametric encoding (hash grids with

learned embeddings) to achieve real-time tracking and high-fidelity surface mapping. ESLAM (13) uses multi-scale feature planes to represent a signed distance field, markedly speeding up the training and inference processes. Other extensions have incorporated higher-level structural mappings, such as object-centric mapping in vMAP (16), which factorizes the scene into individual object latent fields to allow object-level updates. Despite these advances, most neural SLAM systems output *only* geometry and appearance without the corresponding semantic classes.

Haghighi et al. (9) proposed a dense semantic neural SLAM by supervising the implicit map with a 2D segmentation network (limited to a closed label set). (39) incorporated a learned semantic head into the implicit map representation, again assuming a fixed set of semantic classes. In contrast to all the above, our approach *integrates open-vocabulary semantic understanding* into the NICE-SLAM loop, predicting semantics for arbitrary categories on-the-fly alongside camera tracking and mapping.

Open-Vocabulary Semantic Segmentation. Our work is also related to recent progress in open-vocabulary recognition in 2D images. Traditional models are trained on fixed label sets and cannot recognize new classes. To overcome this, open-vocabulary methods leverage vision-language models that connect images to arbitrary text queries. CLIP (23) introduced a powerful image-text embedding space by training on 400 million image-caption pairs. One paradigm generates class-agnostic mask proposals and uses CLIP to classify each mask region as a text label (17). This two-stage design has achieved promising results, although recent work points out that CLIP struggles in masked crops and proposes fine-tuning it on pseudo-mask data to improve segmentation (17). Another line of work trains segmentation models end-to-end with image-text supervision. Ghiasi et al. (8) scale up image-level supervision to learn pixel grouping that aligns with text embeddings. GroupViT (35) learns to segment by grouping visual features with text guidance. We also note the development of segmentation models like Segment Anything (SAM), which produces general mask proposals (15). Such models can be combined with CLIP-based classifiers for open-vocabulary segmentation (17). While open-vocabulary 2D segmentation is now reaching similar performance to closed-set models (17), applying such open-set recognition to 3D mapping raises new challenges. Our approach draws inspiration from these works to encode semantic-rich features, using image-text models to supervise semantics, but we embed this capability into a 3D mapping context.

3D Scene Understanding with Semantics. Bridging multi-view 3D reconstruction with semantics has been explored in both fixed-vocabulary and open-vocabulary settings. Several prior works incorporated semantic labels into implicit 3D representations. Semantic-NeRF (37) extends NeRF by adding an output head for semantic segmentation. However, it relies on a closed label set. OpenScene (21) proposes a zero-shot 3D segmentation approach by aggregating CLIP features from multi-view images into a point cloud. However, a limitation is that the point-wise features in OpenScene do not explicitly separate object instances. To address, OpenMask3D (29) introduces an open-vocabulary 3D *instance* segmentation method. Similarly, ConceptFusion (11) builds an open-set 3D map by combining pixel-aligned features from foundation models into a volumetric map. Jiang et al. (12) use a text-to-image diffusion model and large language models to enrich 3D point features with text descriptors. LERF (14) distills CLIP image embeddings into a radiance field, producing a volumetric language field that can be queried in 3D for arbitrary text labels.

These techniques underscore the growing interest in open-set 3D understanding. However, they typically perform mapping or feature fusion as a *separate* step, requiring pre-scanned scenes (21, 29) or existing reconstruction (11, 14). In contrast, our approach integrates open-set semantic prediction into the neural SLAM pipeline, jointly optimizing camera poses, geometry, and semantic features. This enables an online system that not only reconstructs geometry and appearance like NICE-SLAM, but simultaneously learns an open-vocabulary semantic field.

3 Method

We propose a neural implicit SLAM system that jointly reconstructs geometry, appearance, and semantics of a scene in an online fashion (see Fig. 5). We extend the scene representation of a state-of-the-art neural RGB-D SLAM system (40) to jointly learn a novel open-set semantic feature field and a graph transformer module to enforce context-aware consistency. Our approach learns an implicit 3D map with three components: 1) a geometry field (for occupancy or SDF) that captures the surfaces of the scene, 2) an appearance field for color, and 3) a semantic feature field that encodes high-dimensional descriptors for open-vocabulary semantics. See Figure 2 for a pipeline overview.

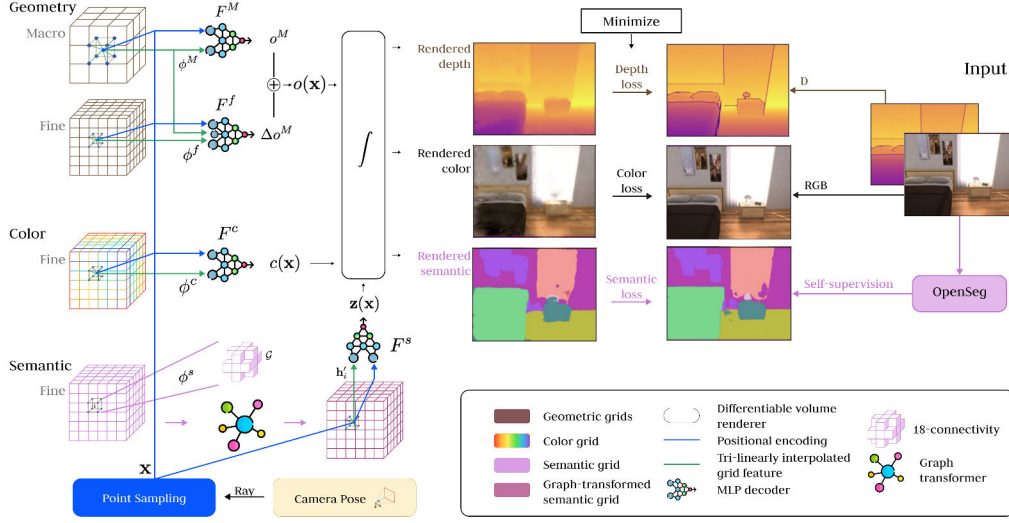


Figure 2: **System Overview.** Our method processes an RGB-D image stream to jointly estimate camera poses and a learned, hierarchical neural scene representation. The pipeline functions as a generative model that synthesizes depth, color, and semantic outputs from the scene representation and camera pose. At test time, both the scene representation – including geometry, appearance, and semantics – and the camera poses are inferred via back-propagation through a differentiable renderer. To train the semantic field, we introduce a novel attention mechanism that aggregates information from neighboring semantic voxels. Estimation is performed via alternating optimization: *Mapping* updates the hierarchical scene representation, semantic and color decoders, and graph transformer parameters, while *Tracking* optimizes the camera pose only.

3.1 3D Scene Representation with Neural Fields

Geometry and Appearance Fields. We represent the scene with a *hierarchical, latent voxel-grid* representation, following NICE-SLAM. In particular, we maintain multi-resolution feature grids that encode the scene geometry and color information on macro and fine scales. Let Φ^g denote the set of geometry feature grids (where $g = \{f, M\}$ standing for fine and macro, respectively) and Φ^c the appearance (color) feature grid. Each feature grid stores a learned feature vector at each voxel, and continuous 3D coordinates $\mathbf{x} \in \mathbb{R}^3$ are processed by trilinear interpolation over the grids to obtain latent codes $\Phi^g(\mathbf{x})$ and $\Phi^c(\mathbf{x})$. These codes are decoded by small MLP networks to predict occupancy and color, respectively. Formally, we define a geometry field $o(\mathbf{x}) \in [0, 1]$ as the occupancy probability at point $\mathbf{x}$. The macro-level occupancy can be computed as follows:

$$o^M(\mathbf{x}) = F^M(\mathbf{x}, \phi^M(\mathbf{x})). \quad (1)$$

where F^M is the macro-level geometry decoder (MLP). Furthermore, the fine-level decoder refines the macro-level occupancy by predicting an offset from the latter based on both the corresponding mid-level feature and the fine-level feature as:

$$\Delta o^M(\mathbf{x}) = F^f(\mathbf{x}, \phi^M(\mathbf{x}), \phi^f(\mathbf{x})), \quad (2)$$

where F^f is the fine-level geometry decoder. Bringing Equations 1 and 2 together, we get:

$$o(\mathbf{x}) = o^M(\mathbf{x}) + \Delta o^M(\mathbf{x}). \quad (3)$$

We also define a color field $c(\mathbf{x})$ as

$$c(\mathbf{x}) = F^c(\mathbf{x}, \phi^c(\mathbf{x})), \quad (4)$$

where F^c is the color decoder. The decoders are lightweight (5 fully connected layers with 32 hidden units), with a skip connection from the input to the third layer to facilitate learning of fine details. Following (40), we also apply a *learnable Gaussian positional encoding* to the coordinate $\mathbf{x}$ before feeding it into the MLPs, which helps the network represent high-frequency details. The hierarchical

grid structure allows macro features to capture larger-scale structure while fine-grid features refine local detail, enabling reconstruction of complex geometry and texture in large scenes.

Volume Rendering for Differentiable Mapping. Similar to NICE-SLAM, we employ differentiable volume rendering to synthesize images from the implicit representation. For each camera ray, a set of sample points $\mathbf{x}_i$ is taken along the ray and queried through the fields to obtain occupancy $o_i = o(\mathbf{x}_i)$ and color $\mathbf{c}_i = \mathbf{c}(\mathbf{x}_i)$. The occupancy values are used to compute the weights for each sample (representing the probability of ray termination at that point). This yields rendered depth $\hat{D}$ and color $\hat{I}$ for the ray by weighted summation of the depth, color, and semantics.

3.2 Open-Set Semantic Feature Field

A key contribution of our approach is the introduction of an *open-set semantic field* that enables semantic mapping beyond a closed set of classes. Instead of predicting discrete labels, we learn a continuous semantic feature representation for each 3D point that can align with a rich vocabulary of concepts. To achieve this, we leverage a pretrained open-vocabulary segmentation model, *OpenSeg* (8), to supervise our 3D semantic features. OpenSeg provides a per-pixel feature vector encoding high-level semantic information (i.e., the embedding produced by the model for that pixel).

We introduce a dedicated voxel grid Φ^s for semantics – at a high resolution that matches the fine geometry grid – and a decoder F^s , i.e. an MLP with architecture similar to the color decoder, that outputs a D -dimensional semantic feature. For any 3D location $\mathbf{x}$, the semantic field produces:

$$\mathbf{z}(\mathbf{x}) = F^s(\mathbf{x}, \phi^s(\mathbf{x})), \quad (5)$$

where $\mathbf{z}(\mathbf{x}) \in \mathbb{R}^D$ lies in the same feature space as OpenSeg’s image embeddings. To be able to supervise this field with the 2D OpenSeg features, we employ the same differentiable rendering procedure explained above, to obtain $\hat{S}$ as a weighted summation of the $\mathbf{z}(\mathbf{x})$ of the samples along a ray. OpenSeg provides an image-derived feature $\mathbf{f}_p$ for the same pixel. We then enforce $\hat{S}_p$ to be similar to $\mathbf{f}_p$ via a cosine similarity loss. Specifically, we minimize:

$$L_{\text{sem}} = \frac{1}{N} \sum_{p=1}^N \left(1 - \frac{\hat{S}_p \cdot \mathbf{f}_p}{\|\hat{S}_p\| \|\mathbf{f}_p\|} \right), \quad (6)$$

summing over all pixel correspondences p , for a total of N pixels uniformly sampled across the frames we are currently considering (more on this in Section 3.4). This encourages $\mathbf{z}_p$ to point in the same direction as $\mathbf{f}_p$ in the feature space. By training the 3D semantic field against the embeddings of a powerful external model, the map can capture arbitrary semantic concepts present in the scene, achieving *open-set semantic* understanding and consistency. In practice, we found that this feature alignment strategy effectively transfers the rich knowledge of the segmentation model into the 3D domain: regions on the map that correspond to, say, ‘chair’ or ‘desk’ get embedded in vectors similar to those of OpenSeg for those objects, even though our system never explicitly defines those classes. In this way, semantic information is retained without being limited to a predefined set of labels.

3.3 Graph Transformer for Contextual Feature Aggregation

Although the semantic feature field provides per-point descriptors, predicting the semantics of each point independently, via an MLP, can lead to inconsistencies or noise, especially in unobserved or ambiguous regions. We address this by introducing a *Graph Transformer* module that operates on the voxel grid to propagate semantic information across space. The idea is to exploit the spatial coherence of real-world scenes: nearby voxels are likely to have a related semantic context (e.g., contiguous surfaces or objects).

We construct a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where each node $i \in \mathcal{V}$ corresponds to a voxel (specifically, a feature in the fine-level grid), and edges $\mathcal{E}$ connect neighboring voxels. Instead of using only the six direct face-adjacent neighbors (as in a 6-connected grid), we use an *18-neighborhood* connectivity: each voxel connects to all voxels that share a face or an edge (but not just a corner). This expanded neighborhood provides a richer local context without the full cost of 26-neighborhood. In our experiments, 6-connectivity was too restrictive (many semantic details were underutilized due to limited context), while 26-connectivity gave only marginal gains at much higher memory usage.

This graph $\mathcal{G}$ is then processed using a graph transformer which is a graph neural network based on multi-head self-attention operating over $\mathcal{G}$ (31). Let $\mathbf{h}_i$ be the initial feature of node i (i.e., $\mathbf{h}_i$

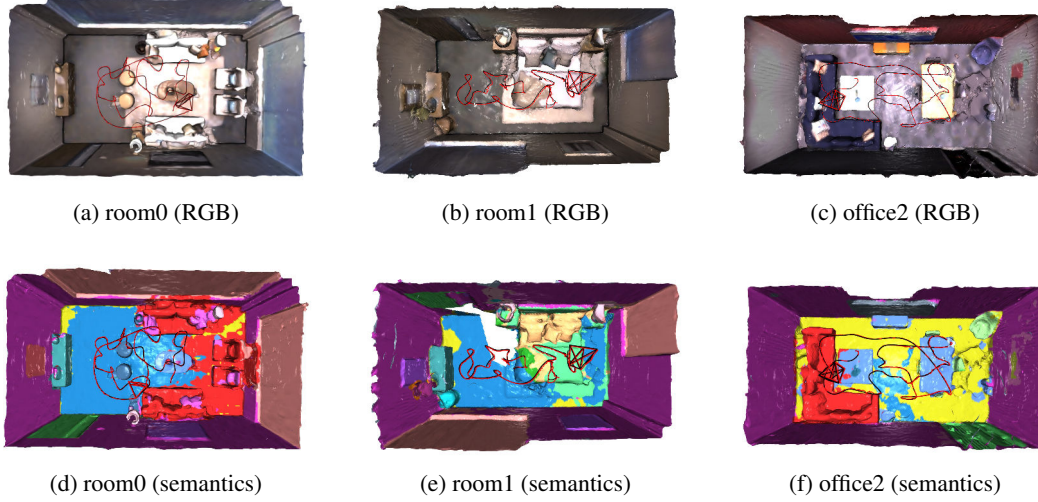


Figure 3: **3D Semantic Reconstructions** by the proposed method and the corresponding estimated camera trajectories on Replica (25). To obtain the multi-class semantic visualizations, we define a list of classes (with corresponding colors) that we are interested in querying and encode these through the CLIP text encoder to obtain the corresponding text features. We then compute the *argmax* of the dot product between those and the visual open-set feature of each pixel, to effectively map the open-vocabulary features to their closest label in the list.

represents the semantic latent code in the voxel updated during training). In one layer of the graph transformer, each node attends to the features of its neighboring nodes to update its representation. In particular, to every node we apply a shared linear transformation parametrized by matrix $\mathbf{W}$, perform the attentional mechanism (denoted with $\mathbf{a}$) to get the pairwise attention coefficients for each neighbor $j \in \mathcal{N}(i)$, where $\mathcal{N}(i)$ is the set of neighbors of i in $\mathcal{G}$, including i itself, and finally normalize them using the softmax function. Formally:

$$\alpha_{ij} = \frac{\exp(\text{LeakyReLU}(\mathbf{a}^T \mathbf{W} \mathbf{h}_i + \mathbf{a}^T \mathbf{W} \mathbf{h}_j))}{\sum_{k \in \mathcal{N}(i)} \exp(\text{LeakyReLU}(\mathbf{a}^T \mathbf{W} \mathbf{h}_i + \mathbf{a}^T \mathbf{W} \mathbf{h}_k))} \quad (7)$$

These coefficients determine how much attention node i pays to each neighbor j . The feature of the node is updated as a weighted sum of the neighbor values as follows:

$$\mathbf{h}'_i = \sigma \left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij} \mathbf{W} \mathbf{h}_j \right). \quad (8)$$

where σ denotes a generic nonlinearity. We employ multi-head attention (with several sets of weight matrices $\mathbf{W}$ learned) to capture different aspects of context, and follow each attention layer with a feed-forward network and layer normalization as in standard Transformers (30). The idea behind this mechanism is to allow information to propagate across the graph, effectively allowing each voxel’s semantics to be influenced by its neighbors. After L layers, the output features $\mathbf{h}_i^{(L)}$ constitute refined semantic embeddings that incorporate spatial context and smoothness. This Graph Transformer module is integrated into our semantic field: the Graph Transformer returns richer features starting from the semantic grid, which are then fed to the MLP F^s .

3.4 Optimization: Tracking and Mapping

We adopt an *alternating tracking-and-mapping* optimization strategy, similar to (40), to jointly estimate camera poses and scene representation. The SLAM system runs two interleaved threads: a *tracking* step that localizes the camera for each incoming frame, and a *mapping* step that updates the implicit model (geometry, appearance, semantics) using the accumulated observations. Both steps rely on the differentiable rendering described above to formulate losses that are minimized via gradient-based optimization.

Camera Tracking. For each new RGB-D frame at time t , we optimize the 6-DoF camera pose $T_w^{c_t}$ so that the rendered view from the current map aligns with the observed image. We fix the map

Table 1: **Open-vocabulary semantic segmentation** performance on the Replica dataset (25). We report mean Intersection over Union (mIoU) and mean Accuracy (mAcc) in percent (%). Our online SLAM method, OpenICE-SLAM (with and without the proposed attention mechanism), is compared against state-of-the-art methods that necessitate pre-existing scene geometry and camera poses (marked as "Offline"). Results are presented for all classes and broken down into head, common, and tail categories. Higher values indicate better performance.

		Tail		Common		Head		All	
		mAcc	mIoU	mAcc	mIoU	mAcc	mIoU	mAcc	mIoU
Offline	LERF (14)	17.3	2.3	31.2	10.1	28.1	19.2	25.8	10.5
	OpenScene (21) (Dis.)	6.7	1.4	21.3	12.8	41.1	30.2	23.0	14.8
	OpenScene (21) (Ens.)	6.3	1.5	22.6	14.5	44.8	31.7	24.6	15.9
	OpenNeRF (7)	17.6	5.8	31.3	20.1	46.2	35.4	31.7	20.4
OpenICE-SLAM w/o Attention		10.0	1.8	30.1	15.1	38.6	23.9	25.9	13.3
OpenICE-SLAM		10.8	7.9	38.4	19.7	40.4	26.3	29.5	15.4

parameters during tracking. Pose optimization minimizes a photometric reprojection error and (if depth is available) depth consistency error. Concretely, given the current estimate of the scene, we render a synthetic color image $\hat{I}_t(T)$ and depth image $\hat{D}_t(T)$ as functions of the candidate pose T . These are compared to the actual camera observations I_t^{obs} and D_t^{obs} . We minimize a weighted sum of an RGB loss (e.g., L_1 or SSIM between $\hat{I}_t$ and I_t^{obs}) and a depth loss (e.g. L_1 between $\hat{D}_t$ and D_t^{obs}). The optimization is done with gradient descent on the pose parameters, utilizing the fact that $\frac{\partial \hat{I}_t}{\partial T}$ and $\frac{\partial \hat{D}_t}{\partial T}$ can be computed through the implicit network and rendering process. We only optimize the pose for the current frame (one variable per frame) during tracking, which keeps this step fast. Once converged, the estimated pose T_w^{ct} is used to incorporate the frame into the map update.

Mapping. The mapping thread continuously optimizes for geometry, appearance and semantics – specifically Φ^g , Φ^c , Φ^s , color and semantic decoders parameters and Graph Transformer parameters – to minimize the reconstruction error across the frames observed so far. Following (40) strategy, we maintain and iteratively update a fixed-size set of keyframes that cover the portion of the scene we are observing, and sample N pixels from these keyframes to evaluate the losses. To ensure a stable and efficient mapping, we divide optimization into stages.

Initially, we optimize only the geometry field using image reconstruction losses, then after 60% of iterations are passed, so once the basic shape of the scene is captured, we introduce the color and semantic fields and Graph Transformer. This staged approach is crucial because the semantic supervision signal can hinder in the early stages of learning geometry. By first optimizing geometry, we obtain a solid backbone model of the scene. Specifically, during this first stage, we minimize $L_{\text{map}}^{\text{base}} = \lambda_d L_{\text{depth}}$, where λ_d is the weighting coefficient and L_{depth} is the L1 loss between the predicted depth $\hat{D}_p$ and the input depth D_p^{obs} for pixel p , i.e., $L_{\text{depth}} = \frac{1}{N} \sum_{p=1}^N |\hat{D}_p - D_p^{\text{obs}}|_1$. This optimizes the voxel grid features Φ^g . Once the geometric reconstruction is sufficiently accurate, we activate the color and semantic learning: in the second stage, after computing OpenSeg features of the selected list of keyframes to self-supervise the semantic predictions, we add L_{sem} as defined earlier and $L_{\text{color}} = \frac{1}{N} \sum_{p=1}^N |\hat{I}_p - I_p^{\text{obs}}|_1$ to the loss, where $\hat{I}_p$ and I_p^{obs} are the predicted and input RGB at point p . The joint mapping loss becomes:

$$L_{\text{map}}^{\text{joint}} = \lambda_d L_{\text{depth}} + \lambda_c L_{\text{color}} + \lambda_s L_{\text{sem}}, \quad (9)$$

where λ_s , λ_c control the influence of the semantic and color terms, respectively. In this phase we are optimizing also for color grid Φ^c , semantic grid Φ^s , color and semantic decoders parameters and Graph Transformer parameters and all components are optimized together to refine the map.

We perform mapping in an online fashion, once every five frames. Gradients are accumulated to update the voxel feature grids and network weights. The Graph Transformer, being integrated into the forward model for semantics, also receives gradients so that it learns to propagate features in ways that reduce the semantic loss. Last, we enable a local bundle adjustment by additionally optimizing $L_{\text{map}}^{\text{joint}}$ over pose parameters T . In practice, our optimization interleaves mapping and tracking processes, enhancing the system to gradually build a coherent map and refine the poses.

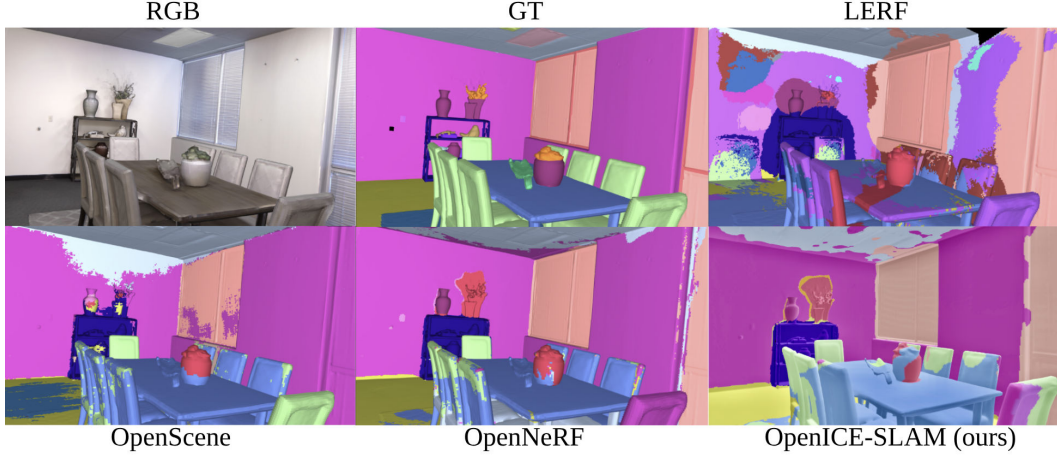


Figure 4: **Qualitative comparison** with offline baselines on 3D open-set segmentation.

4 Experiments

We evaluate the performance of our proposed online open-set neural SLAM system in three key areas: 3D open-vocabulary semantic segmentation, geometric reconstruction quality, and camera tracking accuracy. All experiments are carried out on a PC mounting an AMD Ryzen Threadripper 7960X 24-Cores CPU and a NVIDIA GeForce RTX 4090 GPU.

Datasets. Replica (25) provides high-quality 3D reconstructions of diverse indoor environments with photorealistic textures. It is particularly well-suited for evaluating open-set 3D scene understanding due to its long-tail class distribution and meticulously annotated ground-truth semantic labels, which include very small objects like switches and wall-plugs, unlike other common 3D semantic segmentation datasets such as ScanNet (5) or Matterport3D (4). All our semantic experiments are conducted on the commonly-used 8 scenes: office0, office1, office2, office3, office4, room0, room1, and room2, following prior work (40; 21; 37; 7). Each RGB-D video sequence consists of 2000 frames, downsampled to a resolution of 640×360 pixels. For evaluating camera tracking, we use the widely adopted TUM RGB-D dataset (26). This dataset offers various sequences with ground truth camera trajectories captured by a motion capture system, making it suitable for quantitative evaluation of SLAM systems.

Metrics. Replica is annotated with 51 distinct semantic class categories. For evaluating semantics, we follow (7) and group these categories into three equally sized subsets of 17 classes each: head, common, and tail, based on the number of annotated points per class. Ground-truth semantic labels are used *only* for evaluation. We assess 3D semantic segmentation performance on the reconstructed point clouds using the mean Intersection over Union (mIoU) and mean Accuracy (mAcc) over all ground-truth semantic classes, as well as for each of the head, common, and tail subsets.

We employ both 2D and 3D metrics to evaluate scene geometry. For the 2D metric, we measure the L1 loss on 1000 randomly sampled depth pixels between our rendered depth maps and the ground truth depth maps from the input sequences. For 3D metrics, following standard protocols (6), we report Accuracy in cm as the median distance from reconstructed points to ground truth, Completion in cm, as the median distance from ground truth to reconstruction, and Completion Ratio (< 5cm %) as the percentage of ground truth points within 5cm of the reconstruction. Unseen regions outside of any viewing frustum are excluded. We evaluate camera tracking accuracy using the Absolute Trajectory Error (ATE) Root Mean Squared Error (RMSE) [cm] (26), which measures the global consistency of the estimated trajectory against the ground truth.

Baselines. We compare our semantic accuracy with recent methods that perform 3D semantic understanding with open set capabilities, including LERF (14), OpenScene (21), and OpenNeRF (7). It is important to note that, in contrast to our online SLAM approach, all these baseline methods require pre-estimated scene geometry and accurate camera poses as input or use offline approaches to estimate these, typically operating on already reconstructed scenes.

Table 2: **Geometric Accuracy** on the Replica dataset (25). The proposed OpenICE-SLAM achieves similar accuracy to state-of-the-art while also learning open-set semantics.

	TSDF-Fusion (38)	iMAP (27)	DI-Fusion (10)	NICE-SLAM (40)	OpenICE-SLAM (Ours)
Depth L1 (↓)	7.57	7.64	23.33	3.53	3.67
Acc. (↓)	1.60	6.95	19.40	2.85	3.58
Comp. (↓)	3.49	5.33	10.19	3.00	2.95
Comp. Ratio (↑)	86.08	66.60	72.96	89.33	88.89

Table 3: **Camera Tracking Accuracy**, measured by RMSE (cm) on TUM RGB-D (26). We achieve accuracy on par with other SLAM models, even outperforming some, while also learning semantics.

	fr1/desk	fr2/xyz	fr3/office
iMAP (27)	4.9	2.0	5.8
iMAP* (40)	7.2	2.1	9.0
DI-Fusion (10)	4.4	2.3	15.6
NICE-SLAM (40)	2.7	1.8	3.0
OpenICE-SLAM (Ours)	3.3	2.6	4.2
BAD-SLAM (24)	1.7	1.1	1.7
Kintinuous (34)	3.7	2.9	3.0
ORB-SLAM2 (19)	1.6	0.4	1.0

We compare our geometry and camera poses with dense SLAM systems: traditional methods like TSDF-Fusion (38), and neural implicit SLAM methods such as iMAP (27), DI-Fusion (10), and the foundational NICE-SLAM (40), upon which our geometric and tracking framework builds.

Semantic Accuracy. The results for open-vocabulary 3D segmentation on the Replica dataset are in Table 1. Our complete system (with the graph-transformator attention) demonstrates strong performance in this challenging task. Notably, while OpenNeRF (7), which operates *offline* using pre-existing ground truth geometry and camera poses, reports only a slightly higher mAcc of 31.7%, our OpenICE-SLAM achieves a similar accuracy. Critically, our method performs simultaneous localization, mapping, and semantic segmentation entirely online from raw RGB-D input streams, without any prior knowledge of the scene structure or camera trajectory. This represents a significant practical advantage, as our system builds the semantic map incrementally and robustly online.

The impact of our graph transformer module is clearly evident when comparing our full system to its ablated version, OpenICE-SLAM w/o Attention. The attention mechanism substantially boosts the overall mIoU from 13.3% to 15.4% and mAcc from 25.9% to 29.5%. These results validate that OpenICE-SLAM can generate rich, open-vocabulary semantic maps online, achieving a level of accuracy that approaches offline methods operating under ideal conditions.

Example segmentations, estimated by the baselines and the proposed method, are shown in Fig. 4. OpenICE-SLAM achieves similar segmentation to OpenNeRF, while also estimating the scene geometry and camera trajectories.

Geometry and Camera Tracking. The geometric accuracy on the TUM RGB-D dataset is reported in Table 2. The results show that we achieve similar accuracy to NICE-SLAM, effectively demonstrating that the proposed open-set semantic functionality does not significantly affect other parts of the SLAM system. The camera trajectory errors reported in Table 3 show similar behaviours.

Limitations. Although our system advances accurate *online* open-set 3D semantic reconstruction, it remains slightly behind offline approaches, which benefit from pre-built reconstructions.

5 Conclusion

We introduced OpenICE-SLAM, a neural implicit SLAM framework that jointly reconstructs geometry, appearance, and open-vocabulary semantic labels in an online fashion. By integrating a context-aware semantic feature field with a transformer-based attention mechanism, our method enables consistent and fine-grained semantic understanding during simultaneous localization and mapping. Unlike prior work, OpenICE-SLAM performs all operations online, without requiring

pre-computed reconstructions or known camera poses. Extensive experiments demonstrate that our system achieves competitive semantic accuracy compared to state-of-the-art offline baselines, while maintaining strong geometric reconstruction and camera tracking performance. These results highlight the potential of our approach for real-time open-set scene understanding in practical applications such as robotics, AR, and embodied AI. The code will be made public.

References

- [1] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5855–5864, 2021.
- [2] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5470–5479, 2022.
- [3] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Zip-nerf: Anti-aliased grid-based neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19697–19705, 2023.
- [4] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017.
- [5] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017.
- [6] Angela Dai, Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Christian Theobalt. Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration. *ACM Transactions on Graphics (ToG)*, 36(4):1, 2017.
- [7] Francis Engelmann, Fabian Manhardt, Michael Niemeyer, Keisuke Tateno, Marc Pollefeys, and Federico Tombari. Opennerf: open set 3d neural scene segmentation with pixel-wise features and rendered novel views. *arXiv preprint arXiv:2404.03650*, 2024.
- [8] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 640–657, 2022.
- [9] Yasaman Haghighi, Suryansh Kumar, Jean-Philippe Thiran, and Luc Van Gool. Neural implicit dense semantic slam. *arXiv preprint arXiv:2304.14560*, 2023.
- [10] Jiahui Huang, Shi-Sheng Huang, Haoxuan Song, and Shi-Min Hu. Di-fusion: Online implicit 3d reconstruction with deep priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8932–8941, 2021.
- [11] Krishna Murthy Jatavallabhula, Alihusein Kuwajerwala, Qiao Gu, Mohd Omama, Tao Chen, Alaa Maalouf, Shuang Li, Ganesh Iyer, Soroush Saryazdi, Nikhil Varma Keetha, Ayush Tewari, Joshua B. Tenenbaum, Celso de Melo, Madhava Krishna, Liam Paull, Florian Shkurti, and Antonio Torralba. Conceptfusion: Open-set multimodal 3d mapping. In *Robotics: Science and Systems (RSS)*, 2023.
- [12] Li Jiang, Shaoshuai Shi, and Bernt Schiele. Open-vocabulary 3d semantic segmentation with foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [13] Mohammad Mahdi Johari, Camilla Carta, and François Fleuret. ESLAM: Efficient dense slam system based on hybrid representation of signed distance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17408–17419, 2023.
- [14] Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. LERF: Language embedded radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.
- [15] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023.
- [16] Xin Kong, Shikun Liu, Marwan Taher, and Andrew J. Davison. vMAP: Vectorised object mapping for neural field slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 952–961, 2023.
- [17] Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [18] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Commun. ACM*, 65(1):99–106, 2021.
- [19] Raúl Mur-Artal and Juan D. Tardós. ORB-SLAM2: an open-source SLAM system for monocular, stereo and RGB-D cameras. *IEEE Transactions on Robotics*, 33(5):1255–1262, 2017.
- [20] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5589–5599, 2021.
- [21] Songyou Peng, Kyle Genova, Chiyu Max Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. OpenScene: 3d scene understanding with open vocabularies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–824, 2023.
- [22] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10318–10327, 2021.

- [23] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. *Proceedings of the 38th International Conference on Machine Learning (ICML)*, 139:8748–8763, 2021.
- [24] Thomas Schops, Torsten Sattler, and Marc Pollefeys. Bad slam: Bundle adjusted direct rgb-d slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
- [25] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019.
- [26] Jürgen Sturm, Wolfram Burgard, and Daniel Cremers. Evaluating egomotion and structure-from-motion approaches using the tum rgb-d benchmark. In *Proc. of the Workshop on Color-Depth Camera Fusion in Robotics at the IEEE/RJS International Conference on Intelligent Robot Systems (IROS)*, volume 13, page 6, 2012.
- [27] Edgar Sucar, Shikun Liu, Joseph Ortiz, and Andrew J. Davison. iMAP: Implicit mapping and positioning in real-time. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6229–6238, 2021.
- [28] Ayca Takmaz, Alexandros Delitzas, Robert W. Sumner, Francis Engelmann, Johanna Wald, and Federico Tombari. Search3D: Hierarchical Open-Vocabulary 3D Segmentation. *IEEE Robotics and Automation Letters (RA-L)*, 2025.
- [29] Ayca Takmaz, Elisabetta Fedele, Robert W. Sumner, Marc Pollefeys, Federico Tombari, and Francis Engelmann. OpenMask3D: Open-vocabulary 3d instance segmentation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.
- [30] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- [31] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018.
- [32] Hengyi Wang, Jingwen Wang, and Lourdes Agapito. Co-SLAM: Joint coordinate and sparse parametric encodings for neural real-time slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13293–13302, 2023.
- [33] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. Nerf-: Neural radiance fields without known camera parameters. 2021.
- [34] T. Whelan, M. Kaess, M.F. Fallon, H. Johannsson, J.J. Leonard, and J.B. McDonald. Kintinuous: Spatially extended KinectFusion. In *RSS Workshop on RGB-D: Advanced Reasoning with Depth Cameras*, Sydney, Australia, Jul 2012.
- [35] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18134–18144, 2022.
- [36] Miao Yang, Iro Armeni, Marc Pollefeys, and Daniel Barath. Volumetric semantically consistent 3d panoptic mapping. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2024)*, Abu Dhabi, United Arab Emirates, October 14-18, 2024, 2024.
- [37] Shuaifeng Zhi, Tristan Laidlow, Stefan Leutenegger, and Andrew J. Davison. In-place scene labelling and understanding with implicit scene representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15858–15867, 2021.
- [38] Qian-Yi Zhou and Vladlen Koltun. Dense scene reconstruction with points of interest. *ACM Transactions on Graphics (ToG)*, 32(4):1–8, 2013.
- [39] Siting Zhu, Guangming Wang, Hermann Blum, Jiuming Liu, Liang Song, Marc Pollefeys, and Hesheng Wang. SNI-SLAM: Semantic neural implicit slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [40] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R. Oswald, and Marc Pollefeys. NICE-SLAM: Neural implicit scalable encoding for SLAM. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12786–12796, 2022.

A Additional visualizations

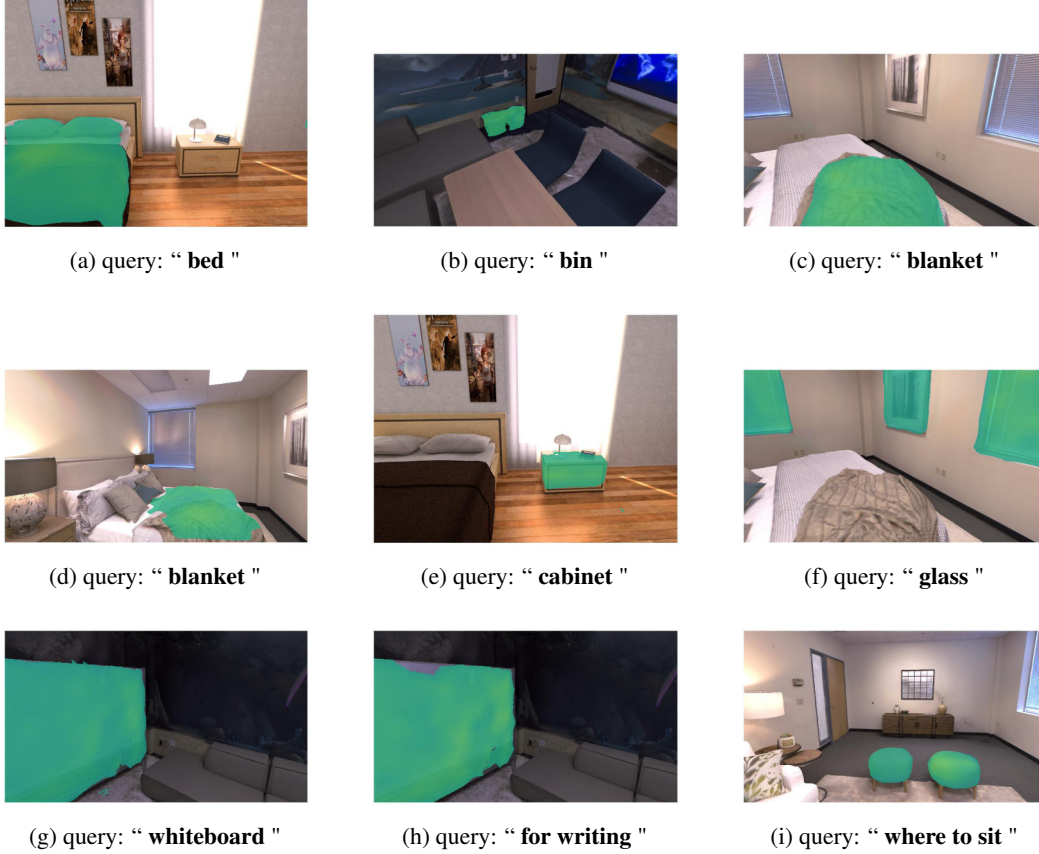


Figure 5: **Visualization of single queries on open-set semantic reconstruction.** Starting from our implicit reconstruction, we render the open-set features for every pixel in the 2D image. We then apply cosine similarity between the embedded query and every pixel to get the per-pixel-probability of belonging to the queried class. For nicer visualizations and clearer understanding, we threshold the probabilities and highlight in green only the most probable pixels on top of the RGB image.

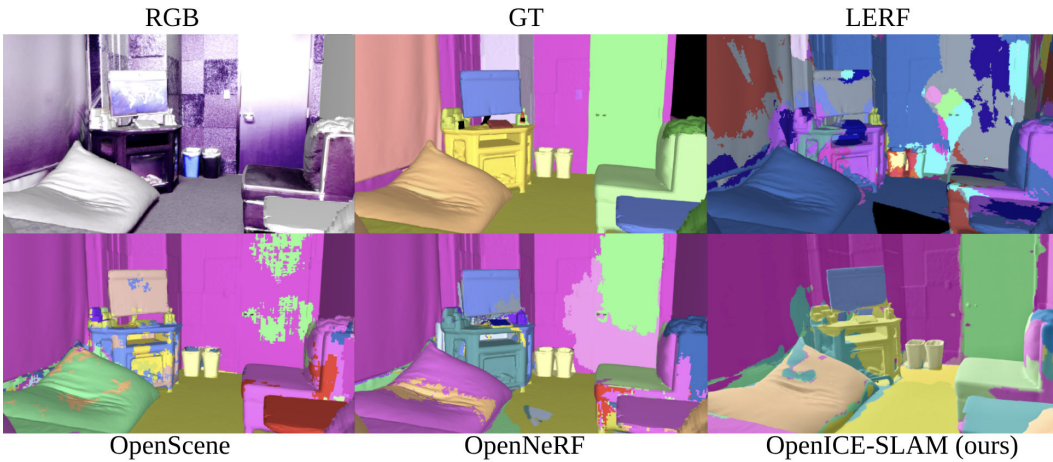


Figure 6